

Amendments In the Claims

Please amend Claims 3, 6, 10, 12-13, 19-21, and 31 as follows:

1-2. Canceled

3. (Currently Amended) A method in a computing system for processing a relational database query, comprising:

forming the relational database query against a relational model of a multidimensional data source using a graphical user interface, wherein the graphical user interface is configured to display a presentation layer representation of a virtual relational table corresponding to the multidimensional data source, enable pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;

receiving the relational database query, the received relational database query being drawn against **the** **[[a]]** relational model of a multidimensional data source, wherein

the relational model comprises a relational-to-multidimensional mapping between **the** **[[a]]** virtual relational table and the multidimensional data source;

using the relational-to-multidimensional mapping together with relational/multidimensional equivalency logic to construct a multidimensional database query based on the received relational database query, wherein

the relational/multidimensional equivalency logic comprises a general mapping between relational queries and structures and multidimensional queries and structures; and

submitting the constructed multidimensional database query for execution against the modeled multidimensional data source.

4. (Original) The method of claim 3 wherein the multidimensional query is constructed in MDX.

5. (Original) The method of claim 4 wherein the relational query is expressed in SQL.

6. (Currently Amended) The method of claim 3, wherein **the graphical user interface is further configured to enable selection of a detail filter to apply against the relational model, and** the relational query specifies the [[a]] detail filter against the relational model, and wherein the constructed multidimensional query specifies that the detail filter be applied to the modeled multidimensional data source.

7. (Original) The method of claim 6 wherein the relational query is expressed in SQL, and wherein the detail filter specified by the relational query is an SQL WHERE clause.

8. (Original) The method of claim 3 wherein the relational query specifies an aggregation function against the relational model, and wherein the constructed multidimensional query specifies that the aggregation function be applied to the modeled multidimensional data source.

9. (Original) The method of claim 8 wherein the relational query is expressed in SQL, and wherein the aggregation function specified by the relational query is an SQL GROUP BY clause.

10. (Currently Amended) The method of claim 3, wherein **the graphical user interface is further configured to enable selection of a summary filter to apply against the relational model,**

the relational query specifies the **[[a]]** summary filter against the relational model,
 and
~~wherein~~ the constructed multidimensional query specifies that the summary filter
 be applied to the modeled multidimensional data source.

11. (Original) The method of claim 10 wherein the relational query is expressed in SQL, and wherein the summary filter specified by the relational query is an SQL HAVING clause.

12. (Currently Amended) The method of claim 3, wherein
the graphical user interface is further configured to enable selection of a
 detail filter to apply against the relational model, and
 the relational query specifies the **[[a]]** detail filter against the relational model
 having selected predicates, and
~~wherein~~ the constructed multidimensional query specifies, for each of the
 selected predicates that can be applied against the modeled
 multidimensional data source before a Crassjoin operation is performed,
 applying the selected predicate against the modeled multidimensional data
 source as early as possible.

13. (Currently Amended) The method of claim 3, wherein
the graphical user interface is further configured to enable selection of a
 detail filter to apply against the relational model, and
 the relational query specifies a detail filter against the relational model having
 selected predicates, and
~~wherein~~ the constructed multidimensional query specifies, for each of the
 selected predicates that can be applied against the modeled
 multidimensional data source before a Crassjoin operation is performed,
 applying the selected predicate against the modeled multidimensional data
 source before the **[[a]]** Crassjoin operation is performed.

14. (Original) The method of claim 3 wherein the relational query specifies performing a selected aggregation function on a selected column of a virtual relational

table, the virtual relational table corresponding to a multidimensional data source, the selected column corresponding to a selected measure of the multidimensional data source, the method further comprising:

retrieving metadata identifying an aggregation function used for the selected measure of the multidimensional data source;
determining whether the aggregation function identified by the metadata matches the selected aggregation function; and
if the aggregation function identified by the metadata matches the selected aggregation function, generating a multidimensional query against the multidimensional data source that relies on the aggregation function performed in the multidimensional data source.

15. (Original) The method of claim 3, further comprising:

receiving, in response to submitting the multidimensional database query, a multidimensional database query result; and
using a relational-to-multidimensional mapping contained by the model together with relational/multidimensional equivalency logic to construct a relational database query result based on the received multidimensional database query result.

16. (Original) The method of claim 3, further comprising:

determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and
constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and
submitting the native relational database query for execution against the conventional relational tables,
and wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source, the method further comprising:

receiving, in response to submitting the native relational database query, a native relational database query result; and
combining the constructed relational database query result with the received native relational database query result in accordance with the received relational database query.

17. (Original) The method of claim 3, further comprising making information about the model available for use in building the received relational database query.

18. (Original) The method of claim 3, further comprising:
determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and
constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and
submitting the native relational database query for execution against the conventional relational tables,
and wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source.

19. (Currently Amended) A computer-readable medium comprising instructions to cause a computing system to process a relational database query, said instructions comprising:

a first set of instructions, executable on a processor, configured to form the relational database query against a relational model of a multidimensional data source using a graphical user interface, wherein the graphical user interface is configured to display a presentation layer representation of a virtual relational table corresponding to the multidimensional data source,

enable pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;

- a **second** ~~first~~ set of instructions, executable on a processor, configured to receive the relational database query, the received relational database query being drawn against **the** ~~[[a]]~~ relational model of **the** ~~[[a]]~~ multidimensional data source, wherein
 - the relational model comprises a relational-to-multidimensional mapping between **the** ~~[[a]]~~ virtual relational table and the multidimensional data source;
- a **third** ~~second~~ set of instructions, executable on the processor, configured to use the relational-to-multidimensional mapping to translate the received relational database query into a multidimensional database query; and
- a **fourth** ~~third~~ set of instructions, executable on the processor, configured to submit the multidimensional database query for execution against the modeled multidimensional data source.

20. (Currently Amended) The computer-readable medium of claim 19 further comprising:

- a **fifth** ~~fourth~~ set of instructions, executable on the processor, configured to receive, in response to submitting the multidimensional database query, a multidimensional database query result; and
- a **sixth** ~~fifth~~ set of instructions, executable on the processor, configured to use a relational-to-multidimensional mapping to translate the received multidimensional database query result into a relational database query result.

21. (Currently Amended) A computing system for processing a relational database query, comprising:

- a graphical user interface subsystem configured to form the relational database query against a relational model of a multidimensional data source, wherein**

the graphical user interface subsystem is further configured to
display a presentation layer representation of a virtual
relational table corresponding to the multidimensional
data source,
enable pointer-driven selection for database query of one or
more tables and columns of data stored in the
multidimensional data source and represented by the
displayed presentation layer;

- a query reception subsystem that receives the relational database query, the received relational database query being drawn against the **[[a]]** relational model of the **[[a]]** multidimensional data source, wherein the relational model comprises a relational-to-multidimensional mapping between the **[[a]]** virtual relational table and the multidimensional data source;
- a multidimensional query construction subsystem that uses the relational-to-multidimensional mapping to construct a multidimensional database query based on the received relational database query; and
- a query submission subsystem that submits the constructed multidimensional database query for execution against the modeled multidimensional data source.

22. (Previously Presented) The computing system of claim 21, further comprising:

- a query result reception subsystem that receives, in response to submitting the multidimensional database query, a multidimensional database query result; and
- a relational query result construction subsystem that uses a relational-to-multidimensional mapping to construct a relational database query result based on the received multidimensional database query result.

23-26. **Canceled.**

27-30. **Canceled.**

31. (Currently Amended) A method in a computing system for processing a relational database query, comprising:

forming the relational database query against a relational model of a multidimensional data source using a graphical user interface, wherein the graphical user interface is configured to display a presentation layer representation of a virtual relational table corresponding to the multidimensional data source, enable pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;

receiving the relational database query, the received relational database query being drawn against both **the** **[[a]]** relational model of a multidimensional data source and a native relational table;

converting the received relational database query into (1) a native relational database query against only the native relational table, and (2) a multidimensional database query against the multidimensional data source;

submitting the native relational database query against the native relational table;

submitting the multidimensional database query against the multidimensional data source; and

combining contents of a first search result produced in response to the native relational database query and a second search result produced in response to the multidimensional database query into a third search result responsive to the received relational database query.

32-34. Canceled